

AMENDMENTS

In the Claims

Claims 33-36 and 71-75 were previously canceled.

Please cancel claims 8 and 9 without prejudice.

Please amend claims 14 and 20 as shown below.

Claims 1-7, 10-32, and 37-70 are pending and are listed following:

1. **(previously presented)** A network system, comprising:
 - a first device to maintain an original resource;
 - a second device to maintain a replica resource remotely from the first device, the replica resource being replicated from the original resource;
 - memory to store a cached descriptor corresponding to the original resource;
 - a security component to determine whether the replica resource will pose a security risk to the second device upon receipt of a request for the replica resource, wherein the request designates a resource locator, the security component:
 - being configured to determine whether the request will pose a security risk to the second device;
 - formulating a descriptor corresponding to the replica resource and comparing the formulated descriptor with the cached descriptor; and
 - if the formulated descriptor and the cached descriptor are not equivalent, formulating a second descriptor corresponding to the original resource and comparing the formulated descriptor with the second descriptor.

1 **2. (original)** A network system as recited in claim 1, wherein the
2 security component determines that the replica resource is not a security risk if the
3 formulated descriptor and the cached descriptor are equivalent.

4
5 **3. (original)** A network system as recited in claim 1, wherein, if the
6 formulated descriptor and the cached descriptor are not equivalent, and if the
7 formulated descriptor and the second descriptor are equivalent, the security
8 component determines that the replica resource is not a security risk.

9
10 **4. (original)** A network system as recited in claim 1, wherein, if the
11 formulated descriptor and the cached descriptor are not equivalent, and if the
12 formulated descriptor and the second descriptor are equivalent, the security
13 component determines that the replica resource is not a security risk, and the
14 cached descriptor is replaced with the second descriptor.

15
16 **5. (original)** A network system as recited in claim 1, wherein, if the
17 formulated descriptor and the cached descriptor are not equivalent, and if the
18 formulated descriptor and the second descriptor are not equivalent, the security
19 component determines that the replica resource is a security risk, and the replica
20 resource is replaced with a copy of the original resource.

1 **6. (original)** A network system as recited in claim 1, wherein, if the
2 formulated descriptor and the cached descriptor are not equivalent, and if the
3 formulated descriptor and the second descriptor are not equivalent, the security
4 component determines that the replica resource is a security risk, the replica
5 resource is replaced with a copy of the original resource, and the cached descriptor
6 is replaced with the second descriptor.

7
8 **7. (original)** A network system as recited in claim 1, wherein the
9 security component formulates the cached descriptor when the original resource is
10 replicated to create the replica resource.

11
12 **8-9. (canceled).**

13
14 **10. (previously presented)** A network system as recited in claim 1,
15 wherein the request further designates the resource locator having a resource path,
16 the resource path identifying a location of the replica resource, and wherein the
17 security component determines that the request is not a security risk if the resource
18 path does not exceed a maximum number of characters.

1 **11. (previously presented)** A network system as recited in claim 1,
2 wherein the request further designates the resource locator having a plurality of
3 arguments, and wherein the security component determines that the request is not
4 a security risk if individual arguments do not exceed a maximum number of
5 characters, and if a total number of characters defining all of the arguments do not
6 exceed a maximum number of characters.

7
8 **12. (previously presented)** A network system as recited in claim 1,
9 wherein the request further designates the resource locator having a resource
10 identifier, and wherein the security component determines that the request is not a
11 security risk if the resource identifier has a valid file extension.

12
13 **13. (previously presented)** A network system as recited in claim 1,
14 wherein:

15 the request further designates the resource locator having a resource path
16 and one or more arguments, the resource path identifying a location of the replica
17 resource and the resource path having a resource identifier;

18 the security component determines that the request is not a security risk if:

19 the resource path does not exceed a maximum number of characters;

20 individual arguments do not exceed a maximum number of
21 characters;

22 a total number of characters defining all of the arguments do not
23 exceed a maximum number of characters; and

24 the resource identifier has a valid file extension.
25

1
2 **14. (currently amended)** A network server, comprising:
3 a server component to receive a request for a resource maintained on the
4 network server and, in response to the request, implement security policies to
5 prevent unauthorized access to the resource;

6 a memory to store a cached descriptor corresponding to the resource; and
7 a security component that is registerable with the server component during
8 run-time to determine whether the request will pose a security risk to the network
9 server, the request posing the security risk if the resource has been corrupted and if
10 execution of the resource will compromise the network server, the security
11 component being configured to:

12 formulate a replica descriptor corresponding to a replica of the
13 resource and compare the replica descriptor with the cached descriptor; and
14 if the replica descriptor and the cached descriptor are not equivalent,
15 formulate a second descriptor corresponding to the resource and compare
16 the replica descriptor with the second descriptor.

17
18 **15. (original)** A network server as recited in claim 14, wherein, if the
19 security component determines that the request will pose a security risk, the
20 security component redirects the request to indicate that the resource is not
21 available.

22
23 **16. (original)** A network server as recited in claim 14, wherein the
24 request designates a resource locator having a resource path, the resource path
25

1 identifying a location of the resource, and wherein the security component
2 determines that the request is not a security risk if the resource path does not
3 exceed a maximum number of characters.

4
5 **17. (original)** A network server as recited in claim 14, wherein the
6 request designates a resource locator having a plurality of arguments, and wherein
7 the security component determines that the request is not a security risk if
8 individual arguments do not exceed a maximum number of characters, and if a
9 total number of characters defining all of the arguments do not exceed a maximum
10 number of characters.

11
12 **18. (original)** A network server as recited in claim 14, wherein the
13 request designates a resource locator having a resource identifier, and wherein the
14 security component determines that the request is not a security risk if the resource
15 identifier has a valid file extension.

1 **19. (original)** A network server as recited in claim 14, wherein:

2 the request designates a resource locator having a resource path and one or
3 more arguments, the resource path identifying a location of the resource and the
4 resource path having a resource identifier;

5 the security component determines that the request is not a security risk if:

6 the resource path does not exceed a maximum number of characters;

7 individual arguments do not exceed a maximum number of
8 characters;

9 a total number of characters defining all of the arguments do not
10 exceed a maximum number of characters; and

11 the resource identifier has a valid file extension.

12
13 **20. (currently amended)** A network server system, comprising:

14 a server component in a network server to receive a request for a resource
15 maintained on the network server, the request designating a resource locator
16 having a resource path that identifies a location of the resource, and, in response to
17 the request, implement security policies to prevent unauthorized access to the
18 resource;

19 a memory to store a cached descriptor corresponding to the resource; and

20 a security component in a computing device remote to the network server
21 and registerable with the server component during run-time to determine whether
22 the resource will pose a security risk to the network server upon receipt of the
23 request, the security component being configured to:

1 formulate a replica descriptor corresponding to a replica of the
2 resource and compare the replica descriptor with the cached descriptor; and
3 if the replica descriptor and the cached descriptor are not equivalent,
4 formulate a second descriptor corresponding to the original resource and
5 compare the replica descriptor with the second descriptor.

6
7 **21. (previously presented)** A network server system as recited in
8 claim 20, wherein, if the security component determines that the resource will
9 pose a security risk, the security component redirects the request to indicate that
10 the resource is not available.

11
12 **22. (previously presented)** A network server system as recited in
13 claim 20, wherein the security component:

14 formulates a descriptor corresponding to the resource;

15 compares the formulated descriptor with a cached descriptor, the cached
16 descriptor corresponding to the resource and formulated when the resource is
17 initially requested; and

18 determines that the resource is not a security risk if the formulated
19 descriptor and the cached descriptor are equivalent.

1 **23. (previously presented)** A network server system as recited in
2 claim 20, wherein the security component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor; and

12 determines that the resource is not a security risk if the formulated
13 descriptor and the second descriptor are equivalent.

1 **24. (previously presented)** A network server system as recited in
2 claim 20, wherein the security component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent,
13 initiates that the resource stored on the network server be replaced with a copy of
14 the original resource maintained on the file server; and

15 initiates that the cached descriptor be replaced with the second descriptor.
16
17
18
19
20
21
22
23
24
25

1 **25. (previously presented)** A network server, comprising:

2 an Internet server to receive a request for a resource maintained on the
3 network server and, in response to the request, implement security policies to
4 prevent unauthorized access to the resource;

5 a security component that is registerable with the Internet server during
6 run-time, the security component having:

7 a validation component to determine whether the request will pose a
8 security risk to the network server by determining if a total number of
9 characters defining all of the arguments of the request exceeds a maximum
10 number of characters; and

11 an integrity verification component to determine whether the
12 resource will pose a security risk to the network server upon receipt of the
13 request.

14
15 **26. (original)** A network server as recited in claim 25, wherein the
16 request designates a resource locator having a resource path, the resource path
17 identifying a location of the resource, and wherein the validation component
18 determines that the request is not a security risk if the resource path does not
19 exceed a maximum number of characters.

1 **27. (previously presented)** A network server as recited in claim 25,
2 wherein the request designates a resource locator having a plurality of arguments,
3 and wherein the validation component determines that the request is not a security
4 risk if individual arguments do not exceed a maximum number of characters.

5
6 **28. (original)** A network server as recited in claim 25, wherein the
7 request designates a resource locator having a resource identifier, and wherein the
8 validation component determines that the request is not a security risk if the
9 resource identifier has a valid file extension.

10
11 **29. (previously presented)** A network server as recited in claim 25,
12 wherein:

13 the request designates a resource locator having a resource path and one or
14 more arguments, the resource path identifying a location of the resource and the
15 resource path having a resource identifier;

16 the validation component determines that the request is not a security risk
17 if:

18 the resource path does not exceed a maximum number of characters;
19 individual arguments do not exceed a maximum number of
20 characters; and

21 the resource identifier has a valid file extension.
22
23
24
25

1 **30. (original)** A network server as recited in claim 25, wherein the
2 integrity verification component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested; and

7 determines that the resource is not a security risk if the formulated
8 descriptor and the cached descriptor are equivalent.

9
10 **31. (original)** A network server as recited in claim 25, wherein the
11 integrity verification component:

12 formulates a descriptor corresponding to the resource;

13 compares the formulated descriptor with a cached descriptor, the cached
14 descriptor corresponding to the resource and formulated when the resource is
15 initially requested;

16 if the formulated descriptor and the cached descriptor are not equivalent,
17 formulates a second descriptor corresponding to an original resource maintained
18 on a file server remotely located from the network server, the resource being
19 replicated from the original resource;

20 compares the formulated descriptor with the second descriptor; and

21 determines that the resource is not a security risk if the formulated
22 descriptor and the second descriptor are equivalent.

1 **32. (original)** A network server as recited in claim 25, wherein the
2 integrity verification component:

3 formulates a descriptor corresponding to the resource;

4 compares the formulated descriptor with a cached descriptor, the cached
5 descriptor corresponding to the resource and formulated when the resource is
6 initially requested;

7 if the formulated descriptor and the cached descriptor are not equivalent,
8 formulates a second descriptor corresponding to an original resource maintained
9 on a file server remotely located from the network server, the resource being
10 replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent,
13 initiates that the resource stored on the network server be replaced with a copy of
14 the original resource maintained on the file server; and

15 initiates that the cached descriptor be replaced with the second descriptor.
16

17 **33-36. canceled**
18
19
20
21
22
23
24
25

1 **37. (previously presented)** One or more computer readable media
2 containing a security application, comprising:

3 a validation component to determine whether a request for a resource poses
4 a security risk by determining if a total number of characters defining all of the
5 arguments of the request exceeds a maximum number of characters; and

6 an integrity verification component to determine whether the resource poses
7 a security risk.

8
9 **38. (original)** Computer readable media as recited in claim 37,
10 wherein the request designates a resource locator having a resource path, the
11 resource path identifying a location of the resource, and wherein the validation
12 component determines that the request is not a security risk if the resource path
13 does not exceed a maximum number of characters.

14
15 **39. (previously presented)** Computer readable media as recited in
16 claim 37, wherein the request designates a resource locator having a plurality of
17 arguments, and wherein the validation component determines that the request is
18 not a security risk if individual arguments do not exceed a maximum number of
19 characters.

1 **40. (original)** Computer readable media as recited in claim 37,
2 wherein the request designates a resource locator having a resource identifier, and
3 wherein the validation component determines that the request is not a security risk
4 if the resource identifier has a valid file extension.

5
6 **41. (previously presented)** Computer readable media as recited in
7 claim 37, wherein:

8 the request designates a resource locator having a resource path and one or
9 more arguments, the resource path identifying a location of the resource and the
10 resource path having a resource identifier;

11 the validation component determines that the request is not a security risk
12 if:

13 the resource path does not exceed a maximum number of characters;
14 individual arguments do not exceed a maximum number of
15 characters; and
16 the resource identifier has a valid file extension.

1 **42. (original)** Computer readable media as recited in claim 37,
2 wherein the integrity verification component:

3 formulates a descriptor corresponding to the resource when the security
4 application receives the request;

5 compares the formulated descriptor with a cached descriptor, the cached
6 descriptor corresponding to the resource and formulated when the resource is
7 initially requested; and

8 determines that the resource is not a security risk if the formulated
9 descriptor and the cached descriptor are equivalent.
10

11 **43. (original)** Computer readable media as recited in claim 37,
12 wherein the integrity verification component:

13 formulates a descriptor corresponding to the resource when the security
14 application receives the request;

15 compares the formulated descriptor with a cached descriptor, the cached
16 descriptor corresponding to the resource and formulated when the resource is
17 initially requested;

18 if the formulated descriptor and the cached descriptor are not equivalent,
19 formulates a second descriptor corresponding to an original resource remotely
20 located, the resource being replicated from the original resource;

21 compares the formulated descriptor with the second descriptor; and

22 determines that the resource is not a security risk if the formulated
23 descriptor and the second descriptor are equivalent.
24
25

1 **44. (original)** Computer readable media as recited in claim 37,
2 wherein the integrity verification component:

3 formulates a descriptor corresponding to the resource when the security
4 application receives the request;

5 compares the formulated descriptor with a cached descriptor, the cached
6 descriptor corresponding to the resource and formulated when the resource is
7 initially requested;

8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulates a second descriptor corresponding to an original resource remotely
10 located, the resource being replicated from the original resource;

11 compares the formulated descriptor with the second descriptor;

12 if the formulated descriptor and the second descriptor are not equivalent;
13 initiates that the resource be replaced with a copy of the original resource; and

14 initiates that the cached descriptor be replaced with the second descriptor.
15
16
17
18
19
20
21
22
23
24
25

1 **45. (previously presented)** A method, comprising:
2 receiving a request for a replica resource stored on a computing device, the
3 request designating a resource locator having a resource path identifying a location
4 of the replica resource;
5 formulating a descriptor corresponding to the replica resource;
6 comparing the formulated descriptor with a cached descriptor
7 corresponding to an original resource stored on a second computing device
8 remotely located from the computing device, the replica resource being replicated
9 from the original resource;
10 determining that the replica resource does not pose a security risk if the
11 formulated descriptor and the cached descriptor are equivalent;
12 if the formulated descriptor and the cached descriptor are not equivalent,
13 formulating a second descriptor corresponding to the original resource;
14 comparing the formulated descriptor with the second descriptor; and
15 determining that the replica resource does not pose a security risk if the
16 formulated descriptor and the second descriptor are equivalent.

17
18 **46. (original)** A method as recited in claim 45, further comprising
19 allowing the request if said determining that the replica resource does not pose a
20 security risk to the computing device.

21
22 **47. (original)** A method as recited in claim 45, further comprising
23 redirecting the request to indicate that the replica resource is not available if
24 determining that the replica resource poses a security risk to the computing device.
25

1
2 **48. (original)** A method as recited in claim 45, further comprising
3 replacing the cached descriptor with the second descriptor if the formulated
4 descriptor and the second descriptor are equivalent.

5
6 **49. (original)** A method as recited in claim 45, further comprising
7 replacing the replica resource with a copy of the original resource if the
8 formulated descriptor and the cached descriptor are not equivalent, and if the
9 formulated descriptor and the second descriptor are not equivalent.

10
11 **50. (original)** A method as recited in claim 45, further comprising
12 replacing the cached descriptor with the second descriptor if the formulated
13 descriptor and the cached descriptor are not equivalent, and if the formulated
14 descriptor and the second descriptor are not equivalent.

15
16 **51. (original)** A method as recited in claim 45, further comprising
17 formulating the cached descriptor when the original resource is replicated to create
18 the replica resource.

19
20 **52. (original)** A method as recited in claim 45, further comprising
21 formulating the cached descriptor when the replica resource is initially requested.

22
23 **53. (original)** A method as recited in claim 45, further comprising
24 determining whether the request will pose a security risk.
25

1
2 **54. (original)** A method as recited in claim 45, further comprising:
3 determining whether the request will pose a security risk; and
4 redirecting the request to indicate that the replica resource is not available if
5 determining that the request poses a security risk to the computing device.

6
7 **55. (previously presented)** A method as recited in claim 45, further
8 comprising determining that the request does not pose a security risk if the
9 resource path does not exceed a maximum number of characters.

10
11 **56. (previously presented)** A method as recited in claim 45, wherein
12 the request further designates the resource locator having a plurality of arguments,
13 and the method further comprising determining that the request does not pose a
14 security risk if individual arguments do not exceed a maximum number of
15 characters, and if a total number of characters defining all of the arguments do not
16 exceed a maximum number of characters.

17
18 **57. (previously presented)** A method as recited in claim 45, wherein
19 the request further designates the resource locator having a resource identifier, and
20 the method further comprising determining that the request does not pose a
21 security risk if the resource identifier has a valid file extension.

1 **58. (previously presented)** A method as recited in claim 45,
2 wherein:

3 the request further designates the resource locator having a resource path
4 and one or more arguments, the resource path identifying a location of the replica
5 resource and the resource path having a resource identifier;

6 the method further comprising determining that the request does not pose a
7 security risk if:

8 the resource path does not exceed a maximum number of characters;
9 individual arguments do not exceed a maximum number of
10 characters;

11 a total number of characters defining all of the arguments do not
12 exceed a maximum number of characters; and

13 the resource identifier has a valid file extension.

14
15 **59. (original)** A computer-readable medium comprising computer
16 executable instructions that, when executed, direct a computing system to perform
17 the method of claim 45.

18
19 **60. (original)** A computer-readable medium comprising computer
20 executable instructions that, when executed, direct a computing system to perform
21 the method of claim 58.

1 **61. (previously presented)** A method, comprising:
2 receiving a request for a resource;
3 implementing security policies to prevent unauthorized access to the
4 resource;
5 determining whether the request will pose a security risk by determining if
6 a total number of characters defining all of the arguments of the request exceeds a
7 maximum number of characters; and
8 determining whether the resource will pose a security risk if allowing the
9 request.

10
11 **62. (original)** A method as recited in claim 61, further comprising
12 allowing the request for the resource if determining that the request does not pose
13 a security risk and if determining that the resource does not pose a security risk.

14
15 **63. (original)** A method as recited in claim 61, wherein the request
16 designates a resource locator having a resource path, the resource path identifying
17 a location of the resource, and the method further comprising determining that the
18 request does not pose a security risk if the resource path does not exceed a
19 maximum number of characters.

1 **64. (previously presented)** A method as recited in claim 61, wherein
2 the request designates a resource locator having a plurality of arguments, and the
3 method further comprising determining that the request does not pose a security
4 risk if individual arguments do not exceed a maximum number of characters.

5
6 **65. (original)** A method as recited in claim 61, wherein the request
7 designates a resource locator having a resource identifier, and the method further
8 comprising determining that the request does not pose a security risk if the
9 resource identifier has a valid file extension.

10
11 **66. (original)** A method as recited in claim 61, further comprising:
12 formulating a descriptor corresponding to the resource;
13 comparing the formulated descriptor with a cached descriptor
14 corresponding to the resource and formulated when the resource is initially
15 requested; and
16 determining that the resource does not pose a security risk if the formulated
17 descriptor and the cached descriptor are equivalent.

1 **67. (original)** A method as recited in claim 61, further comprising:
2 formulating a descriptor corresponding to the resource;
3 comparing the formulated descriptor with a cached descriptor
4 corresponding to the resource and formulated when the resource is initially
5 requested;
6 determining that the resource does not pose a security risk if the formulated
7 descriptor and the cached descriptor are equivalent;
8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulating a second descriptor corresponding to an original resource remotely
10 located, the resource replicated from the original source;
11 comparing the formulated descriptor with the second descriptor; and
12 determining that the resource does not pose a security risk if the formulated
13 descriptor and the second descriptor are equivalent.

1 **68. (original)** A method as recited in claim 61, further comprising:
2 formulating a descriptor corresponding to the resource;
3 comparing the formulated descriptor with a cached descriptor
4 corresponding to the resource and formulated when the resource is initially
5 requested;
6 determining that the resource does not pose a security risk if the formulated
7 descriptor and the cached descriptor are equivalent;
8 if the formulated descriptor and the cached descriptor are not equivalent,
9 formulating a second descriptor corresponding to an original resource remotely
10 located, the resource replicated from the original resource;
11 comparing the formulated descriptor with the second descriptor; and
12 determining that the resource does not pose a security risk if the formulated
13 descriptor and the second descriptor are equivalent;
14 if the formulated descriptor and the second descriptor are not equivalent,
15 replacing the resource with a copy of the original resource and replacing the
16 cached descriptor with the second descriptor.

17
18 **69. (original)** A computer-readable medium comprising computer
19 executable instructions that, when executed, direct a computing system to perform
20 the method of claim 61.

21
22 **70. (original)** A computer-readable medium comprising computer
23 executable instructions that, when executed, direct a computing system to perform
24 the method of claim 68.
25

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

71-75. canceled